Weakening and Iterating Laws Using String Diagrams

Alexandre Goy[1,2]

LIP
ENS Lyon
France

Abstract

Distributive laws are a standard way of combining two monads, providing a compositional approach for reasoning about computational effects in semantics. Situations where no such law exists can sometimes be handled by weakening the notion of distributive law, still recovering a composite monad. A celebrated result from Eugenia Cheng shows that combining n monads is possible by iterating more distributive laws, provided they satisfy a coherence condition called the Yang-Baxter equation. Moreover, the order of composition does not matter, leading to a form of associativity.
The main contribution of this paper is to generalise the associativity of iterated composition to weak distributive laws in the case of $n = 3$ monads. To this end, we use string-diagrammatic notation, which significantly helps make increasingly complex proofs more readable. We also provide examples of new weak distributive laws arising from iteration.

Keywords: monad, distributive law, weak distributive law, iterated distributive law, Yang-Baxter equation, string diagram

1 Introduction

Monads have been very successful in computer science to model computational effects [29,32,37]. In this context, being able to compose monads is of central importance. The primary framework for monad composition is undoubtedly Beck's celebrated theory of distributive laws [1]. In practice, finding distributive laws between monads is hard, although some recipes are known to build them [26,27]. In contrast, once a law is found, it often contains a significant semantic content – thus, it is common to find papers entirely dedicated to describing the construction and the consequences of a distributive law, see *e.g.* [21]. More generally, monad composition is still a very active field in the computer science community nowadays, with possible applications in coalgebra [34,22,40,5] or in the semantics of computation [14,30].

With the help of a distributive law, one can compose two monads. Composing more monads can be performed by *iterating* [11], *i.e.*, by using more distributive laws – yet, these must meet coherence axioms called the *Yang-Baxter* equations. While distributive laws are the first step towards monad compositionality, *iterated* distributive laws are all the following steps.

Although category theory is a language of choice to work on compositionality properties, there is one major obstacle in the field of distributive laws: monads may not always compose. And consequently,

---

distributive laws may not always exist. The literature exhibits many such negative results, called *no-go theorems* [36,25,13,41,33,42]. A possible fix consists in *weakening* the notion of distributive law to recover transformations that, even if not satisfying all the usual axioms, still enable some weak kind of monad composition. This simple idea has received substantial coverage in the computer science community since the paper of Garner [15]. Simplifying earlier work from [35,9,10], Garner proposed to delete the one axiom that was obstructing, in many concrete cases, the existence of a law. Several papers followed on this revived notion of *weak distributive law* [17,18,4,31]. In general, finding examples of weak distributive laws remains difficult, and very few techniques are known to produce them. Moreover, for such a law to be relevant, it needs to fit into the sweet spot where no distributive law can exist, but there is still enough structure for a weak one. This situation arises in a handful of known cases, *e.g.*, when distributing the distribution monad over the powerset monad [17] or the powerset monad over itself [15,18].

Iterating weak distributive laws is a promising target: one would enable building new weak distributive laws out of some existing ones at low cost. However, two factors are now interacting with the basic theory of Beck. The proof of the iteration theorem [11, Appendix A], due to Eugenia Cheng, is not very difficult per se, but, already for the simplest case of three monads, it contains large commutative diagrams that are hard to comprehend in one glimpse. On the other hand, although weak distributive laws are conceptually close to distributive laws, they make the framework grow in complexity. Manipulating weak laws involves manipulating more natural transformations arising from splitting of idempotents, making proofs less transparent. In short, *iterating* adds one layer of complexity, *weakening* adds another layer, and computations become untractable using the traditional tools of category theory. For a similar situation, consider the paper of Winter [38] whose main result is to prove a variation of Cheng's theorem in the case of three monads. In a nutshell, the variation consists in modifying the type of *one* of the laws from $TF \Rightarrow FT$ into $TF \Rightarrow FST$ – with, of course, consequences for the coherence conditions. Despite this seemingly mild modification, one of the required commutative diagrams becomes so large that finding the appropriate tiling required Winter to use automation via a Prolog program, mainly because of the explosion in the number of naturality squares. As he insists on, the main difficulty is about finding the proof – verifying it remains straightforward.

In the present paper, we aim to illustrate yet another case where *string diagrams* bring conceptual clarity and simplicity. Under the name string diagrams, there are many graphical calculi based on topological operations such as string bending, dragging, and sliding. Such frameworks have raised more and more interest over the past years: we can mention, *e.g.*, the book of Coecke and Kissinger [12] and the line of work of Bonchi, Zanasi *et al.* [39,8,7,6,3,2]. String diagrams are nowadays recognised as a helpful tool. Inside published papers, commutative diagrams have long been preferred, even when they are less enlighting (see *e.g.* [35, Propositions 2.3, 3.3, and 4.1]). One of the aims of this paper is to emphasise the following point: when it comes to concepts that rely on several complexity layers, such as *iterating* plus *weakening*, string diagrams may even be *required* as a computation-assisting technology. The string diagrams we will make use of are those introduced in the style of Hinze and Marsden [28,20,19]. These were already used to produce new results within the theory of distributive laws by Zwart [41] and in the author's PhD thesis [16]. There seems to be also some work in progress concerning no-go theorems [24], showing that string diagrams are imposing themselves in the area to visualise categorical transformations, provide elegant proofs, and communicate ideas.

Contributions.

This paper contains contributions of two kinds: technical (laws) and practical (strings). On the practical side, the paper introduces in Section 3 a convenient string-diagrammatic notation that helps manipulate concepts at play. In particular, we introduce a new notation for idempotents of the form $\kappa\colon ST \Rightarrow ST$ in equation (12), which is central in the whole development. The main technical contribution of the paper consists in generalising Cheng's Theorem 4.1 to weak distributive laws in the case of $n = 3$ monads. For clarity, this result is split into three distinct statements: Theorem 4.3, Theorem 4.5, and Theorem 4.6. Along the way, and building on a key technical tool which is Lemma 3.10, we derive several results of the theory of weak distributive laws, some of them already presented in [16], some of them unpublished. In Section 4.2, we also provide four families of examples where our main result applies:

trivial weak distributive laws, and generic distributive laws involving the exception monad, the reader monad, and the writer monad. Finally, in Section 4.3, we give a new perspective on algebras for a monad by identifying them as distributive laws (Proposition 4.16). In this vision, algebras for a distributive law are identified as iterated distributive laws (Proposition 4.17).

Related Work.

The content of the present paper is partly based on the author's PhD thesis [16], with a slightly different perspective. In [16] the focus is put on the correspondence weak distributive laws - weak liftings - weak extensions. In contrast, in the present paper, we instead follow the approach of [35, Section 2] consisting in building the weak composite monad directly out of the weak distributive law. It has to be mentioned that iterated distributive laws have already been studied from a string-diagrammatic perspective in a paper from Hinze and Marsden [19].

2 Background

2.1 Preliminaries and Notation

Definitional equality is denoted by $\triangleq$. We assume basic knowledge of category theory, including: category, functor, natural transformation. Let $\mathcal{C}$ be a category and S, T be endofunctors on $\mathcal{C}$. We use the Lie bracket notation $[S, T]$ as a shortcut for the type of natural transformations $TS \Rightarrow ST$. For example, $\lambda \colon [S, T]$ means that λ is a natural transformation of type $TS \Rightarrow ST$. This notation is convenient because (a) laws $\lambda : [S, T]$ are related to composite monads $S \circ T$ and (b) the type $TS \Rightarrow ST$ looks like a logical version of the formal expression $-TS + ST$. Identity morphisms will be denoted by $1_X \colon X \to X$, the identity functor by $1 \colon \mathcal{C} \to \mathcal{C}$, and the identity natural transformation on a functor F by $1_F \colon F \Rightarrow F$.

In the whole paper we assume that $\mathcal{C}$ is idempotent complete, that is, for every morphism $k \colon X \to X$ in $\mathcal{C}$ such that $k \circ k = k$, there are morphisms $p \colon X \to Y$ and $i \colon Y \to X$ such that $i \circ p = k$ and $p \circ i = 1_Y$. A direct consequence is that for any functor $F \colon \mathcal{C} \to \mathcal{C}$, every idempotent natural transformation $\kappa \colon F \Rightarrow F$ splits. Explicitly, there are a functor K and natural transformations $\pi \colon F \Rightarrow K$, $\iota \colon K \Rightarrow F$ such that $\iota \circ \pi = \kappa$ and $\pi \circ \iota = 1_K$. All the forecoming examples live in the category Set of sets and functions, which is idempotent complete. So are, *e.g.*, toposes and the category of compact Hausdorff spaces [18]. We globally impose idempotent completeness to be able to split natural transformations denoted by κ in the sequel – all our results remain valid by only asserting locally that these κ split.

2.2 Introducing String Diagrams

We strongly encourage the reader to refer to [20, Sections 2 and 3] for a thorough introduction to string diagrams. In this short section, we will nonetheless provide a crash course in string diagrams, mainly to fix notations.

The following example sums up all useful notation. On the left part, we provide several basic natural transformations $\lambda \colon TS \Rightarrow ST$, $\nu \colon ST \Rightarrow U$, $\varepsilon \colon U \Rightarrow 1$, $\eta \colon 1 \Rightarrow T$. On the right part, we provide two equivalent string diagrams representing the same composite natural transformation $\nu \circ \lambda \varepsilon \circ \eta SU = \nu \circ \lambda \circ \eta S \circ S\varepsilon \colon SU \Rightarrow U$.

We use the following convention: diagrams are read from right to left and from bottom to top (note that [20] rather uses top to bottom). A functor $F \colon \mathcal{C} \to \mathcal{D}$, represented by a vertical string, delineates two regions: the region on the right represents the category $\mathcal{C}$, while the region on the left represents

the category $\mathcal{D}$. In what follows, we only study endofunctors in $\mathcal{C}$ [3]. Therefore all delineated regions remain blank and represent the category $\mathcal{C}$. The identity functor $1\colon \mathcal{C} \to \mathcal{C}$ is not depicted. A natural transformation $\alpha\colon F \Rightarrow G$ is depicted by a symbol between functor F (below) and functor G (above). Using colours, we will define on-the-fly unambiguous symbols (typically, nodes) to distinguish between different natural transformations – see equation (1) for a first example. The identity natural transformation $1_F\colon F \Rightarrow F$ is a string with no symbol, *i.e.*, it coincides with the representation of the functor $F\colon \mathcal{C} \to \mathcal{C}$.

Let F, G, H, K be functors, and $\alpha\colon F \Rightarrow G$, $\beta\colon G \Rightarrow H$, $\gamma\colon H \Rightarrow K$ be natural transformations. Composition of functors is denoted by GF and depicted by horizontal juxtaposition of strings. Vertical composition of natural transformations is denoted by $\beta \circ \alpha\colon F \Rightarrow H$, defined by $(\beta \circ \alpha)_X \triangleq \beta_X \circ \alpha_X$ for all objects X, and depicted by vertical glueing of string diagrams. Left (respectively right) composition of a functor and a natural transformation is denoted by $H\alpha\colon HF \Rightarrow HG$ (respectively $\alpha H\colon FH \Rightarrow GH$), defined by $(H\alpha)_X \triangleq H(\alpha_X)$ (respectively $(\alpha H)_X \triangleq \alpha_{HX}$) for all objects X, and depicted by horizontal juxtaposition of string diagrams. Horizontal composition of natural transformations is denoted by $\gamma\alpha\colon HF \Rightarrow KG$, defined by $\gamma\alpha \triangleq K\alpha \circ \gamma F = \gamma G \circ H\alpha$, and depicted by horizontal juxtaposition of string diagrams.

Each string diagram represents a natural transformation of some type. Two string diagrams of the same type are identified up to continuous operations such as dragging nodes and bending strings (provided it does not reverse the implicit bottom-up vertical direction). Discontinuous operations include sliding a node past another on the same string, crossing two strings, and deleting or adding strings or nodes. Such operations may be allowed punctually in the presence of a natural transformation that justifies them. For instance, a natural transformation $\lambda\colon [S, T]$, having type $TS \Rightarrow ST$, represents an *ad hoc* rule to cross the two strings representing S and T. String-diagrammatic notation is coherent with the usual laws of category theory in the sense that two identified string diagrams always denote the same natural transformation.

It is customary, as done in [20], to specify the types of the functors below and above each string diagram. In this paper, only a few functors are in play, so that our colour code would be sufficient to infer all types. For the sake of clarity, we opt for some redundancy by always indicating types.
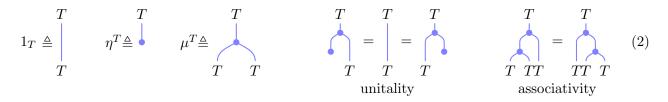
3   A String-Diagrammatic Theory of Weak Distributive Laws

The present section introduces weak distributive laws in string-diagrammatic style, starting back from the standard theory of monads and distributive laws.

3.1   *Monads and Distributive Laws*

Definition 3.1 A *monad* is a triple (T, η^T, μ^T) comprising a $\mathcal{C}$-endofunctor T and two natural transformations, the *unit* $\eta^T\colon 1 \Rightarrow T$ and the *multiplication* $\mu^T\colon TT \Rightarrow T$, such that $\mu^T \circ \eta^T T = 1_T = \mu^T \circ T\eta^T$ (unitality) and $\mu^T \circ \mu^T T = \mu^T \circ T\mu^T$ (associativity).

String diagrammatically, these data are denoted by

(2)

For the sake of readability, we may denote a monad (T, η^T, μ^T) simply by T. Let us give some examples of monads in the category Set.

---

[3] To be precise, we are using the standard formalism of string diagrams for monoidal categories [23] within the category of $\mathcal{C}$-endofunctors, where monoidal product is given by functor composition.

Example 3.2 The *identity* monad consists of the identity functor 1 with both the unit and the multiplication being the identity natural transformation $1 \Rightarrow 1$.

Example 3.3 The *exception monad* E is defined as follows. Let $1 = \{*\}$ be a singleton set. For any set X, let $\mathsf{inl}_X\colon X \to X + 1$ and $\mathsf{inr}_X\colon 1 \to X + 1$ be the canonical injections. The functor E maps a set X to the set $EX \triangleq X + 1$ and a function $f\colon X \to Y$ to $Ef\colon X + 1 \to Y + 1$, defined by $Ef(\mathsf{inl}_X(x)) \triangleq \mathsf{inl}_Y(f(x))$ and $Ef(\mathsf{inr}_X(*)) \triangleq \mathsf{inr}_Y(*)$. The unit is $\eta^E \triangleq \mathsf{inl}\colon 1 \Rightarrow E$ and the multiplication $\mu^E\colon EE \Rightarrow E$ is given by merging exceptions, *i.e.*, $\mu^E_X(\mathsf{inl}_{X+1}(z)) \triangleq z$ for all $z \in X + 1$ and $\mu^E_X(\mathsf{inr}_{X+1}(*)) \triangleq \mathsf{inr}_X(*)$.

Example 3.4 The *reader monad* R is defined as follows. Let A be a set of labels. The functor R maps a set X to the set $RX \triangleq X^A$ and a function $f\colon X \to Y$ to $Rf\colon X^A \to Y^A$, defined by $Rf(h) \triangleq f \circ h$ for all $h \in X^A$. The unit $\eta^R\colon 1 \Rightarrow R$ produces a constant function $\eta^R_X(x) \triangleq (a \mapsto x)$. The multiplication $\mu^R\colon RR \Rightarrow R$ duplicates the input, *i.e.*, $\mu^R_X(h) \triangleq (a \mapsto h(a)(a))$ for all $h \in (X^A)^A$.

Example 3.5 The *writer monad* W is defined as follows. Let $(M, \cdot, e)$ be a monoid. The functor W maps a set X to the set $WX \triangleq M \times X$ and a function $f\colon X \to Y$ to $Wf\colon M \times X \to M \times Y$, defined by $Wf(m, x) \triangleq (m, f(x))$. The unit $\eta^W\colon 1 \Rightarrow W$ outputs the monoid unit $\eta^W_X(x) \triangleq (e, x)$. The multiplication $\mu^W\colon WW \Rightarrow W$ implements the monoid multiplication, *i.e.*, $\mu^W_X(m, (n, x)) \triangleq (m \cdot n, x)$.

Example 3.6 The *powerset monad* P is defined as follows. The functor P maps a set X to the set of its subsets PX and a function $f\colon X \to Y$ to its direct image $Pf\colon PX \to PY$, defined by $Pf(U) \triangleq \{f(x) \mid x \in U\}$ for all $U \subseteq X$. The unit $\eta^P\colon 1 \Rightarrow P$ produces singletons $\eta^P_X(x) \triangleq \{x\}$. The multiplication $\mu^P\colon PP \Rightarrow P$ computes unions, *i.e.*, $\mu^P_X(\mathcal{U}) \triangleq \bigcup \mathcal{U}$ for all $\mathcal{U} \in PPX$.

Example 3.7 The *distribution monad* D is defined as follows. The functor D maps a set X to the set of all finitely-supported probability distributions on X, *i.e.*,

$$DX \triangleq \left\{ p\colon X \to [0,1] \mid \sum_{x \in X} p(x) = 1, \ \mathsf{supp}_X(p) \text{ is finite} \right\} \tag{3}$$

where $\mathsf{supp}_X(p) = \{x \in X \mid p(x) > 0\}$. For a $p \in DX$, we may use the formal sum notation $p = \sum_{x \in X} p_x \cdot x$, where $p_x \triangleq p(x)$. A function $f\colon X \to Y$ is mapped to $Df\colon DX \to DY$, defined by $Df(\sum_{x \in X} p_x \cdot x) \triangleq \sum_{x \in X} p_x \cdot f(x)$. The unit $\eta^D\colon 1 \Rightarrow D$ produces Dirac distributions $\eta^D_X(x) \triangleq 1 \cdot x$. The multiplication $\mu^D\colon DD \Rightarrow D$ is a weighted average, defined by $\mu^D_X(q) \triangleq \sum_{p \in DX} q_p p_x \cdot x$ for all $q \in DDX$.

Definition 3.8 Given two monads S, T on $\mathcal{C}$, a *distributive law* is a $\lambda\colon [S,T]$ such that the four following axioms hold:
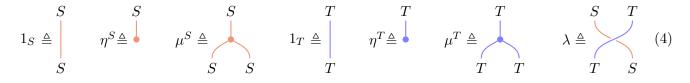
$$\lambda \circ T\eta^S = \eta^S T \tag{η^+}$$
$$\lambda \circ T\mu^S = \mu^S T \circ S\lambda \circ \lambda S \tag{μ^+}$$
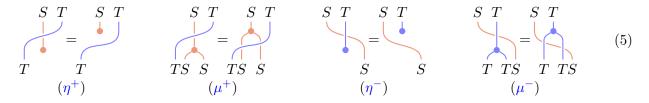$$\lambda \circ \eta^T S = S\eta^T \tag{η^-}$$
$$\lambda \circ \mu^T S = S\mu^T \circ \lambda T \circ T\lambda \tag{μ^-}$$

String diagrammatically, the monads S, T, and the distributive law $\lambda\colon [S,T]$ are pictured as

and the distributive law axioms are

$$
\tag{5}
$$

Any distributive law $\lambda \colon [S,T]$ yields a monad $S \circ_\lambda T \triangleq (ST, \eta^S \eta^T, \mu^S \mu^T \circ S\lambda T)$, also denoted by $S \circ T$ when no confusion can arise, and pictured as

$$
S \circ T \triangleq \qquad \eta^{S \circ T} \triangleq \qquad \mu^{S \circ T} \triangleq
\tag{6}
$$

3.2   Weak Distributive Laws

Definition 3.9 Given two monads S, T on $\mathcal{C}$, a *weak distributive law* is a $\lambda \colon [S,T]$ such that the axioms (η^+), (μ^+) and (μ^-) hold.

In this definition, the weakening comes from dropping the (η^-) axiom, as done in [15]. There are other possible weakenings that we briefly discuss in Remark 3.15. An interesting natural transformation emerges from the structure of a weak distributive law. Consider

$$
\kappa \triangleq \;\; ST \xRightarrow{\eta^T ST} TST \xRightarrow{\lambda T} STT \xRightarrow{S\mu^T} ST
\tag{7}
$$

If λ were a distributive law, then κ would be the identity. In the weak case though, κ still is an idempotent that is compatible with units and multiplications in the following sense.

Lemma 3.10 *The natural transformation κ satisfies the following equations.*

$$
\kappa \circ \kappa = \kappa
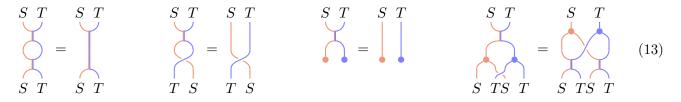\tag{8}
$$
$$
\kappa \circ \lambda = \lambda
\tag{9}
$$
$$
\kappa \circ \eta^S \eta^T = \eta^S \eta^T
\tag{10}
$$
$$
\kappa \circ \mu^S \mu^T \circ S\lambda T = \mu^S \mu^T \circ S\lambda T \circ \kappa\kappa
\tag{11}
$$

As $\kappa \colon ST \Rightarrow ST$ is an idempotent natural transformation and we assume $\mathcal{C}$ to be idempotent complete, there are a $\mathcal{C}$-endofunctor K and two natural transformations $\pi \colon ST \Rightarrow K$ and $\iota \colon K \Rightarrow ST$ such that $\iota \circ \pi = \kappa$ and $\pi \circ \iota = 1_K$. Using these, one can build a monad that is to be interpreted as a weak composition of S and T, namely $S \bullet_\lambda T = (K, \pi \circ \eta^S \eta^T, \pi \circ \mu^S \mu^T \circ S\lambda T \circ \iota\iota)$, also denoted by $S \bullet T$ when the context is clear. See Appendix A for a proof that this is indeed a monad. We now introduce some string diagrammatic notation that will make computations practicable:

$$
1_K \triangleq \qquad \pi \triangleq \qquad \iota \triangleq \qquad = \qquad =
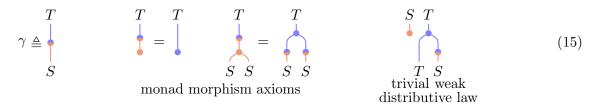\tag{12}
$$

String diagrammatically, the equations from Lemma 3.10 can be represented as below.

$$
\tag{13}
$$

The composite monad $S \bullet T$ is, in turn, as follows – compare with diagrams (6) for $S \circ T$:

$$
1_{S\bullet T} \triangleq \qquad \eta^{S\bullet T} \triangleq \qquad \mu^{S\bullet T} \triangleq \tag{14}
$$

Example 3.11 Let $\gamma \colon S \Rightarrow T$ be a monad morphism, that is, a natural such that $\gamma \circ \eta^S = \eta^T$ and $\gamma \circ \mu^S = \mu^T \circ \gamma\gamma$. Then $\eta^S \mu^T \circ T\gamma \colon [S, T]$ is a weak distributive law – and, in general, not a distributive law. Such distributive laws are called *trivial*, because $S \bullet T = T$ [16, Proposition 2.12]. In diagrams:

$$
\gamma \triangleq \qquad \qquad = \qquad \qquad = \tag{15}
$$

monad morphism axioms

trivial weak
distributive law

We provide two concrete examples:

- The identity monad morphism $1_P \colon P \Rightarrow P$ yields a trivial weak distributive law $\eta^P \mu^P \colon [P, P]$ defined by $\mathcal{U} \mapsto \{\bigcup \mathcal{U}\}$ for any subset of subsets $\mathcal{U}$.

- The support monad morphism $\mathsf{supp} \colon D \Rightarrow P$ yields a trivial weak distributive law $\eta^D \mu^P \circ P\mathsf{supp} \colon [D, P]$ defined by $U \mapsto 1 \cdot \bigcup_{p \in U} \mathsf{supp}_X(p)$ for any subset of distributions $U \in PDX$.

Example 3.12 Given a monad T and a natural transformation $\alpha \colon T \Rightarrow T$, one can view α as having type $[1, T]$ or $[T, 1]$, where 1 is the identity monad described in Example 3.2. Table 1 describes how axioms of distributive laws can then be satisfied by α. One can infer that $\alpha \colon [1, T]$ is a weak distributive law if

Table 1
Required axioms for the law α

axiom	$\alpha \colon [1, T]$	$\alpha \colon [T, 1]$
(η^+)	$\alpha = 1$	$\alpha \circ \eta^T = \eta^T$
(η^-)	$\alpha \circ \eta^T = \eta^T$	$\alpha = 1$
(μ^+)	$\alpha = \alpha \circ \alpha$	$\alpha \circ \mu^T = \mu^T \circ \alpha\alpha$
(μ^-)	$\alpha \circ \mu^T = \mu^T \circ \alpha\alpha$	$\alpha = \alpha \circ \alpha$

and only if α is the identity, whereas $\alpha \colon [T, 1]$ is a weak distributive law exactly when α is an idempotent monad morphism. For example, given any monad morphism $\gamma \colon T \Rightarrow 1$ (that is, a natural transformation which is pointwise Eilenberg-Moore), the corresponding trivial weak distributive law is $\eta^T \circ \gamma \colon [T, 1]$ which is indeed an idempotent monad morphism. We provide two concrete examples:

- The map $\mathsf{proj}^2\colon W \Rightarrow 1$ from the writer monad, defined by $\mathsf{proj}^2_X(m,x) \triangleq x$ for any $(m,x) \in WX$, is a monad morphism. Therefore $\eta^W \circ \mathsf{proj}^2\colon [W,1]$, defined by $(m,x) \mapsto (e,x)$, is a weak distributive law.

- Given any letter $a \in A$, the map $\hat{a}\colon R \Rightarrow 1$ from the reader monad, defined by $\hat{a}_X(h) \triangleq h(a)$ for any $h \in RX$, is a monad morphism. Therefore $\eta^R \circ \hat{a}\colon [R,1]$, defined by $h \mapsto (b \mapsto h(a))$, is a weak distributive law.

We provide two non-trivial examples of weak distributive laws, both of them arising from situations when no distributive law of their type exists [25,36].

Example 3.13 The natural transformation $\lambda\colon [P,D]$ defined for any distribution of subsets $p = \sum_i p_i \cdot U_i$ by

$$\lambda_X(p) = \left\{ \sum_i p_i q^i \mid q^i \in DX, \mathsf{supp}(q^i) \subseteq U_i \right\} \tag{16}$$

is the unique monotone weak distributive law of its type – see [17] for a detailed study.

Example 3.14 The natural transformation $\lambda\colon [P,P]$ defined for any subset of subsets $\mathcal{U}$ by

$$\lambda_X(\mathcal{U}) = \left\{ V \subseteq \bigcup \mathcal{U} \mid \forall U \in \mathcal{U}, V \cap U \neq \emptyset \right\} \tag{17}$$

is the unique monotone weak distributive law of its type – see [15,18] for more details.

One can see that the theory of weak distributive laws is very similar to Beck's theory of distributive laws [1] *modulo* the transformations π and ι. For example, if $\lambda\colon [S,T]$ is a distributive law, then the classical theory shows that $S\eta^T\colon S \Rightarrow S \circ_\lambda T$ and $\eta^S T\colon T \Rightarrow S \circ_\lambda T$ are monad morphisms. Similarly, if $\lambda\colon [S,T]$ is only weak, then $\pi \circ S\eta^T\colon S \Rightarrow S \bullet_\lambda T$ and $\pi \circ \eta^S T\colon T \Rightarrow S \bullet_\lambda T$ are monad morphisms. Whenever the base category is idempotent complete, as is assumed in this paper, the renowned correspondence distributive laws - extensions - liftings also generalises (see [15]).

Remark 3.15 Deleting the (η^-) axiom is one of the many ways of weakening the notion of distributive law. In recent years, this approach has been very fruitful for computer science applications [17,18,16,4]. One may ask, symmetrically, what happens by deleting the (η^+) axiom instead. This idea gives rise to the notion of *coweak distributive law* developed by the author in [16]. Coweak distributive laws enjoy many symmetries with respect to weak distributive laws, but no non-trivial examples of them are known. Yet another way of weakening the notion of distributive law is the one of Street [35], who suggests keeping a weakened version of both the (η^+) and the (η^-) axiom. The presentation of the present paper – including the string diagrammatic approach – is greatly inspired by the one of Street-weak distributive laws in [35]. A broader, 2-categorical account of distributive law weakenings can be found in [9,10].

4 Iterating Laws

4.1 Adapting Cheng's Theorem

In this section, we are interested in what happens when combining more laws. For the sake of simplicity, we will only consider the case when three laws are in play. In the whole section, let R, S, T be monads

$$\sigma T \circ S\tau \circ \lambda R = R\lambda \circ \tau S \circ T\sigma \tag{YB}$$

and $\lambda\colon [S,T]$, $\sigma\colon [R,S]$, $\tau\colon [R,T]$ be natural transformations such that the following equation holds.

Equation (YB), known as the *Yang-Baxter equation*, intuitively states that these three natural transformations are coherent with each other. Diagrammatically, it means that when performing the three crossings in a row, their order does not matter:

$$
\lambda \triangleq \begin{array}{c} S \quad T \\ \reflectbox{X} \\ T \quad S \end{array} \qquad
\sigma \triangleq \begin{array}{c} R \quad S \\ X \\ S \quad R \end{array} \qquad
\tau \triangleq \begin{array}{c} R \quad T \\ X \\ T \quad R \end{array} \qquad
\begin{array}{c} RS \quad T \\ X \\ TS \quad R \end{array} = \begin{array}{c} R \quad ST \\ X \\ T \quad SR \end{array}
\tag{19}
$$

Yang-Baxter

The result of Cheng [11, Theorem 1.6] can be rephrased as follows in the case of three monads.

Theorem 4.1 (Cheng) *Let R, S, T be monads and $\lambda\colon [S,T]$, $\sigma\colon [R,S]$, $\tau\colon [R,T]$ be distributive laws satisfying the Yang-Baxter equation. Then*

- $R\lambda \circ \tau S\colon [R \circ S, T]$ *is a distributive law,*
- $\sigma T \circ S\tau\colon [R, S \circ T]$ *is a distributive law,*
- *these two laws generate the same monad, i.e., $(R \circ S) \circ T = R \circ (S \circ T)$.*

The result can be generalised to weak distributive laws. Let us consider separately what happens for the two possible composite laws. From now onwards, the natural transformations κ, π and ι from Section 3 will mention explicitly which law they relate to – *e.g.*, given a weak distributive law $\lambda\colon [S,T]$, they are denoted by κ^λ, π^λ and ι^λ.

4.1.1 First Composite Law

Assume $\sigma\colon [R,S]$ is a weak distributive law. We will derive which axioms on λ and τ are sufficient for

$$
\varphi \triangleq \; T(R \bullet S) \xRightarrow{T\iota^\sigma} TRS \xRightarrow{\tau S} RTS \xRightarrow{R\lambda} RST \xRightarrow{\pi^\sigma T} (R \bullet S)T
\tag{20}
$$

to be a (weak) distributive law. The natural $\varphi\colon [R \bullet S, T]$ can be depicted as follows, where H is short notation for the functor $R \bullet S$:

$$
\varphi \triangleq \begin{array}{c} H \;\; T \\ \text{(diagram)} \\ T \;\; H \end{array}
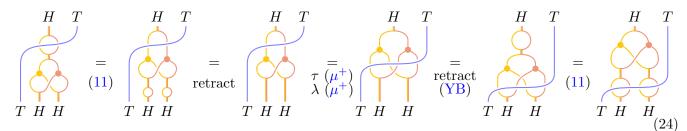\tag{21}
$$

The (η^+) axiom for φ can be derived from equation (10) which states compatibility between units and the idempotent, the (η^+) axiom for τ, and the (η^+) axiom for λ:

$$
\begin{array}{cccc}
\begin{array}{c} H \;\; T \\ \text{(diag)} \\ T \end{array} &
\underset{(10)}{=}
\begin{array}{c} H \;\; T \\ \text{(diag)} \\ T \end{array} &
\underset{\tau\,(\eta^+)}{=}
\begin{array}{c} H \;\; T \\ \text{(diag)} \\ T \end{array} &
\underset{\lambda\,(\eta^+)}{=}
\begin{array}{c} H \;\; T \\ \text{(diag)} \\ T \end{array}
\end{array}
\tag{22}
$$

The (η^-) axiom for φ can be derived from the (η^-) axiom for τ, the (η^-) axiom for λ, and the retract equation $\pi \circ \iota = 1_{R\bullet S}$:

$$
\begin{array}{cccc}
\begin{array}{c} H \;\; T \\ \text{(diag)} \\ H \end{array} &
\underset{\tau\,(\eta^-)}{=}
\begin{array}{c} H \;\; T \\ \text{(diag)} \\ H \end{array} &
\underset{\lambda\,(\eta^-)}{=}
\begin{array}{c} H \;\; T \\ \text{(diag)} \\ H \end{array} &
\underset{\text{retract}}{=}
\begin{array}{c} H\,T \\ \text{(diag)} \\ H \end{array}
\end{array}
\tag{23}
$$

The (μ^+) axiom for φ can be derived from equation (11) which states compatibility between multiplications and the idempotent, the retract equation $\pi \circ \iota = 1_{R\bullet S}$, the (μ^+) axiom for both τ and λ, the Yang-Baxter

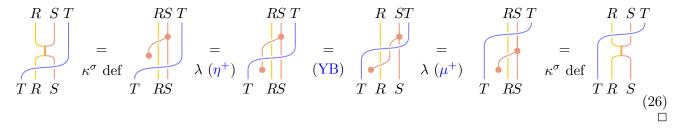equation, the retract equation again, and equation (11) again:

$$(24)$$

To prove the last axiom, we need a technical lemma.

Lemma 4.2 *If λ's axioms (η^+) and (μ^+) hold, and the Yang-Baxter equation holds, then the idempotent κ^σ commutes with the natural $R\lambda \circ \tau S$ in the sense that*

$$\kappa^\sigma T \circ R\lambda \circ \tau S = R\lambda \circ \tau S \circ T\kappa^\sigma \qquad (25)$$

Proof. Easy to see using string diagrams:

$$(26)$$

$\square$

The (μ^-) axiom for φ can then be derived from the retract equation $\pi \circ \iota = 1_{R\bullet S}$, the (μ^-) axiom for both τ and λ, and the lemma relying on λ's axioms (η^+) and (μ^+).

$$(27)$$

Table 2 sums up which axioms of τ and λ are sufficient to make axioms hold for φ.

Table 2
Required axioms for the composite law φ

φ	requires	τ	λ
(η^+)		(η^+)	(η^+)
(η^-)		(η^-)	(η^-)
(μ^+)		(μ^+)	(μ^+)
(μ^-)		(μ^-)	$(\mu^-), (\eta^+), (\mu^+)$

Using the information contained in Table 2, we get the following result.

Theorem 4.3 *Let R, S, T be monads and $\lambda \colon [S, T]$, $\sigma \colon [R, S]$, $\tau \colon [R, T]$ be weak distributive laws satisfying the Yang-Baxter equation. Then the composite $\varphi \triangleq \pi^\sigma T \circ R\lambda \circ \tau S \circ T\iota^\sigma \colon [R \bullet S, T]$ is a weak distributive law. If, additionally, τ and λ are distributive laws, then φ is a distributive law.*

4.1.2 Second Composite Law

One could also compose the laws the other way around. Assume $\lambda\colon [S,T]$ is a weak distributive law. Consider the natural transformation

$$\psi \triangleq (S \bullet T)R \xRightarrow{\iota^\lambda R} STR \xRightarrow{S\tau} SRT \xRightarrow{\sigma T} RST \xRightarrow{R\pi^\lambda} R(S \bullet T) \tag{28}$$

The natural $\psi\colon [R, S \bullet T]$ can be depicted as follows, where K is short notation for the functor $S \bullet T$:

$$\psi \triangleq \quad \tag{29}$$

Again, assuming some axioms of distributive laws for σ and τ, one can infer axioms for ψ. Using string diagrams, one can easily see that the proofs are symmetrical from the ones in Section 4.1.1. As an instance, recall that Lemma 4.2 was required to prove the (μ^-) axiom for φ. Now, the following symmetrical result is required to prove the (μ^+) axiom for ψ:

Lemma 4.4 *If τ's axioms (η^-) and (μ^-) hold, and the Yang-Baxter equation holds, then the idempotent κ^λ commutes with the natural $\sigma T \circ S\tau$ in the sense that*

$$R\kappa^\lambda \circ \sigma T \circ S\tau = \sigma T \circ S\tau \circ \kappa^\lambda R \tag{30}$$
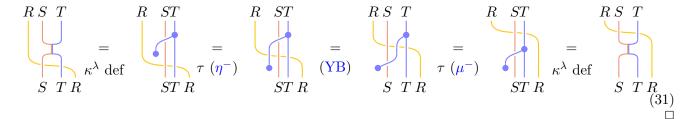
Proof. Easy to see using string diagrams:

Table 3 sums up which axioms of σ and τ are sufficient to make axioms hold for ψ.

Table 3
Required axioms for the composite law ψ

ψ	requires	σ	τ
(η^+)		(η^+)	(η^+)
(η^-)		(η^-)	(η^-)
(μ^+)		(μ^+)	(μ^+), (η^-), (μ^-)
(μ^-)		(μ^-)	(μ^-)

Using the information contained in Table 3, we get the following result.

Theorem 4.5 *Let R, S, T be monads, $\lambda\colon [S,T]$, $\sigma\colon [R,S]$ be weak distributive laws, and $\tau\colon [R,T]$ be a distributive law, satisfying the Yang-Baxter equation. Then the composite $\psi \triangleq R\pi^\lambda \circ \sigma T \circ S\tau \circ \iota^\lambda R\colon [R, S\bullet T]$ is a weak distributive law. If, additionally, σ is a distributive law, then ψ is a distributive law.*

4.1.3 Associativity of Weak Iteration

According to the preceding paragraphs, provided the Yang-Baxter equation holds and enough axioms are satisfied, there are two ways of obtaining a composite weak distributive law. The first one, described in Section 4.1.1, yields by Theorem 4.3 a weak distributive law $\varphi\colon [R \bullet S, T]$ and thus a monad $(R \bullet S) \bullet T$. The second one, described in Section 4.1.2, yields by Theorem 4.5 a weak distributive law $\psi\colon [R, S \bullet T]$ and thus a monad $R \bullet (S \bullet T)$. Contrary to Theorem 4.1, it is not obvious that even the underlying functors of these two monads are the same. It turns out that they are, and that more broadly, one can identify these two monads, allowing us to write unambiguously $R \bullet S \bullet T$.
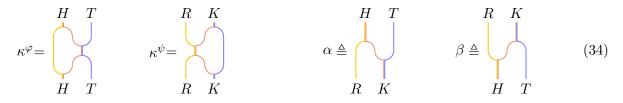
Theorem 4.6 *Let R, S, T be monads, $\lambda\colon [S,T]$, $\sigma\colon [R,S]$ be weak distributive laws and $\tau\colon [R,T]$ be a distributive law, satisfying the Yang-Baxter equation. Then the weak distributive laws $\varphi \triangleq \pi^\sigma T \circ R\lambda \circ \tau S \circ T\iota^\sigma\colon [R\bullet S, T]$ and $\psi \triangleq R\pi^\lambda \circ \sigma T \circ S\tau \circ \iota^\lambda R\colon [R, S\bullet T]$ generate the same monad, i.e., $(R \bullet S)\bullet T = R\bullet(S \bullet T)$.*

Proof. After some string-diagrammatic computations (see Appendix B), one can express the idempotents κ^φ and κ^ψ as follows:

$$\kappa^\varphi = \pi^\sigma T \circ R\kappa^\lambda \circ \iota^\sigma T = \alpha \circ \beta \tag{32}$$

$$\kappa^\psi = R\pi^\lambda \circ \kappa^\sigma T \circ R\iota^\lambda = \beta \circ \alpha \tag{33}$$

where $\alpha \triangleq \pi^\sigma T \circ R\iota^\lambda$ and $\beta \triangleq R\pi^\lambda \circ \iota^\sigma T$, as pictured below.

$$\tag{34}$$

Now let us split $\kappa^\varphi = \iota^\varphi \circ \pi^\varphi$ with $\pi^\varphi \circ \iota^\varphi = 1_{(R\bullet S)\bullet T}$. We show that this splitting generates also a splitting for κ^ψ. Indeed, one can choose $\iota^\psi \triangleq \beta \circ \iota^\varphi$ and $\pi^\psi \triangleq \pi^\varphi \circ \alpha$ because

$$\iota^\psi \circ \pi^\psi = \beta \circ \iota^\varphi \circ \pi^\varphi \circ \alpha = \beta \circ \kappa^\varphi \circ \alpha = \beta \circ \alpha \circ \beta \circ \alpha = \kappa^\psi \circ \kappa^\psi = \kappa^\psi \tag{35}$$

$$\pi^\psi \circ \iota^\psi = \pi^\varphi \circ \alpha \circ \beta \circ \iota^\varphi = \pi^\varphi \circ \kappa^\varphi \circ \iota^\varphi = (\pi^\varphi \circ \iota^\varphi) \circ (\pi^\varphi \circ \iota^\varphi) = 1_{(R\bullet S)\bullet T} \tag{36}$$

Hence the splitting of κ^ψ is as follows:

$$\tag{37}$$
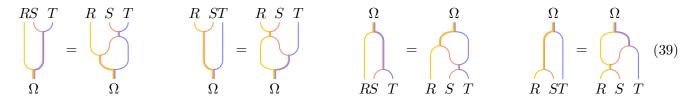
Therefore, the functors $(R \bullet S) \bullet T$ and $R \bullet (S \bullet T)$ can be identified. We denote it by $R \bullet S \bullet T$ (short notation Ω) and picture it as follows:

$$\tag{38}$$

Moreover, one can separate the three functors – or merge the three functors – in any order *up to an*

idempotent, as shown in the following diagrams (proved in Appendix B):

$$\tag{39}$$

Using these compatibility properties along with aforementioned properties of κ, it is then a matter of lengthy but straightforward computations to show that units and multiplications of $(R \bullet S) \bullet T$ and $R \bullet (S \bullet T)$ coincide (see Appendix B). □

4.2   Examples

In this section, we illustrate many cases where the results of the previous section can be applied by proving the Yang-Baxter equation for some usual monads and (weak) distributive laws.

4.2.1   Iterating with Trivial Weak Distributive Laws

When one of the three laws involved is trivial, the Yang-Baxter equation usually simplifies to a condition relating the underlying monad morphism to the other laws. We mention the two following cases.

Proposition 4.7 *Let $\tau \colon [R,T]$ be a weak distributive law.*

- *Let $\sigma \colon [R,S]$ be a distributive law and $\lambda \triangleq \eta^S \mu^T \circ T\gamma \colon [S,T]$ be a trivial weak distributive law arising from a monad morphism $\gamma \colon S \Rightarrow T$ such that $R\gamma \circ \sigma = \tau \circ \gamma R$. Then the Yang-Baxter equation is satisfied.*

- *Let $\lambda \colon [S,T]$ be a weak distributive law and $\sigma \triangleq \eta^R \mu^S \circ S\gamma \colon [R,S]$ be a trivial weak distributive law arising from a monad morphism $\gamma \colon R \Rightarrow S$ such that $\gamma T \circ \tau = \lambda \circ \gamma T$. Then the Yang-Baxter equation is satisfied.*

4.2.2   Iterating with Exceptions

The exception monad E from Example 3.3 can be described by seeing $\mathsf{inl} \colon 1 \Rightarrow E$ and $\mathsf{inr} \colon \underline{1} \Rightarrow E$ as natural transformations, where 1 is the identity functor and $\underline{1}$ is the constant functor $\underline{1}X \triangleq 1 = \{*\}$, $\underline{1}f \triangleq 1_{\{*\}}$. One can then define a well-known generic distributive law relying directly on these two coproduct injections.

Example 4.8 For any Set monad T, there is a distributive law $\varepsilon^T \colon [T, E]$ defined by

$$\varepsilon^T \circ \mathsf{inl}T = T\eta^E \tag{40}$$
$$\varepsilon^T \circ \mathsf{inr}T = \eta^T E \circ \mathsf{inr} \tag{41}$$

Moreover, for monads S and T, the distributive laws ε^S and ε^T are coherent with each other in the sense of the next proposition.

Proposition 4.9 *Let $\lambda \colon [S,T]$ be a natural transformation such that either the (η^+) or the (η^-) axiom holds. Then the following Yang-Baxter equation holds:*

$$\lambda E \circ T\varepsilon^S \circ \varepsilon^T S = S\varepsilon^T \circ \varepsilon^S T \circ E\lambda \tag{42}$$

Remark 4.10 The above proposition still holds true when E implements multiple exceptions. More generally, it can be generalised to any category with binary coproducts and any exception monad of the form $EX = X + e$, where e is an object of exceptions.

Consequently, by applying Theorem 4.6, given any weak distributive law $[S,T]$, one can safely compose the resulting monad with the exception monad E and get a monad $S \bullet T \circ E$. We give two explicit examples.

- Applying Theorem 4.5 to the weak distributive law $\lambda\colon [P, D]$ from Example 3.13, we derive a new weak distributive law $\lambda E \circ D\varepsilon^P\colon [P, D_\leq]$ between the powerset monad P and the *subdistribution monad* $D_\leq = D \circ E$. The expression of this new law is as in (16).
- Remarking that $P = P_* \circ E$ where P_* is the *non-empty* powerset monad, we identify the weak distributive law $\lambda\colon [P, P]$ from Example 3.14 as an iterated law built from ε^P, ε^{P_*}, and a weak distributive law of type $[P, P_*]$ whose expression is as in (17).

4.2.3 Iterating with the Reader

Recall that A is a set of labels and the reader monad in Set is defined as $RX = X^A$. For any $a \in A$, define the function $\hat{a}_X\colon RX \to X$ by $\hat{a}_X(h) = h(a)$.

Example 4.11 For any Set monad T, there is a distributive law $\rho^T\colon [R, T]$ defined for any set X and any $z \in TRX$ by

$$\rho^T_X(z) \triangleq a \mapsto T\hat{a}_X(z) \tag{43}$$

Moreover, for monads S and T, the distributive laws ρ^S and ρ^T are coherent with each other in the sense of the next proposition.

Proposition 4.12 Let $\lambda\colon [S, T]$ be a natural transformation. Then the following Yang-Baxter equation holds:

$$\rho^S T \circ S\rho^T \circ \lambda R = R\lambda \circ \rho^T S \circ T\rho^S \tag{44}$$

Applying Theorem 4.6 with *e.g.* the weak $\lambda\colon [P, D]$ from Example 3.13, we deduce that the monad $R \circ P \bullet D$ can be obtained either using the distributive law $\psi = \rho^{P \bullet D}\colon [R, P \bullet D]$ or using the weak distributive law $\varphi \triangleq R\lambda \circ \rho^D P\colon [R \circ P, D]$ given by the expression

$$\varphi_X\left(\sum_{h \in (PX)^A} p_h \cdot h\right) = a \mapsto \left\{\sum_{h \in (PX)^A} p_h q^h \mid q^h \in DX,\, \mathsf{supp}(q^h) \subseteq h(a)\right\} \tag{45}$$

4.2.4 Iterating with the Writer

Recall that M is a monoid and the writer monad in Set is defined as $MX = M \times X$. For any $m \in M$, let $\overline{m}_X\colon X \to M \times X$ be defined by $\overline{m}_X(x) = (m, x)$.

Example 4.13 For any Set monad T, there is a distributive law $\omega^T\colon [T, W]$ defined for any set X and any $(m, t) \in WTX$ by

$$\omega^T_X(m, t) \triangleq T\overline{m}_X(t) \tag{46}$$

Moreover, for monads S and T, the distributive laws ω^S and ω^T are coherent with each other in the sense of the next proposition.

Proposition 4.14 Let $\lambda\colon [S, T]$ be a natural transformation. Then the following Yang-Baxter equation holds:

$$\lambda W \circ T\omega^S \circ \omega^T S = S\omega^T \circ \omega^S T \circ W\lambda \tag{47}$$

Applying Theorem 4.6 with *e.g.* the weak $\lambda\colon [P, D]$ from Example 3.13, we deduce that the monad $P \bullet D \circ W$ can be obtained either using the distributive law $\varphi = \omega^{P \bullet D}\colon [P \bullet D, W]$ or using the weak distributive law $\psi \triangleq \lambda W \circ T\omega^S\colon [P, D \circ W]$ given by the expression

$$\psi_X\left(\sum_{i,j} p_{i,j} \cdot (m_i, U_j)\right) = \left\{\sum_{i,j} p_{i,j} q^{i,j} \mid q^{i,j} \in DWX,\, \mathsf{supp}(q^{i,j}) \subseteq \{(m_i, x) \mid x \in U_j\}\right\} \tag{48}$$

4.3 Algebras as Distributive Laws, and Iterating them

In this short section we sketch how Eilenberg-Moore algebras are a particular kind of distributive laws, and how in this view λ-algebras are precisely Yang-Baxter equations. Given a monad T, an *Eilenberg-Moore algebra* – or *T-algebra* for short – is a pair (A, t) where A is an object of $\mathcal{C}$ and $t\colon TA \to A$ is a morphism such that $t \circ \eta_A^T = 1_A$ (unitality) and $t \circ \mu_A^T = t \circ Tt$ (associativity). A morphism of T-algebras $f\colon (A, t) \to (B, u)$ is a morphism $f\colon A \to B$ in $\mathcal{C}$ such that $f \circ t = u \circ Tf$. For any functor $F\colon \mathcal{C} \to \mathcal{C}$, an *Eilenberg-Moore distributive law* – or *EM-law* for short – is a natural transformation $\lambda\colon [F, T]$ such that the (η^-) and (μ^-) axioms are satisfied. Let S and T be two monads and $\lambda\colon [S, T]$ be a natural transformation. A *λ-algebra* is a triple (A, s, t) such that (A, s) is an S-algebra, (A, t) is a T-algebra, and the following condition is satisfied:

$$s \circ St \circ \lambda_X = t \circ Ts \qquad\qquad (\lambda\text{-condition})$$

A morphism of λ-algebras is a simultaneous S-algebra and T-algebra morphism.

The following result is widely known:

Theorem 4.15 ([1,15]) *Let $\lambda\colon [S, T]$ be a natural transformation.*

- *If λ is a distributive law, then the categories of $(S \circ_\lambda T)$-algebras and λ-algebras are isomorphic.*
- *If λ is a weak distributive law, then the categories of $(S \bullet_\lambda T)$-algebras and λ-algebras are isomorphic.*

Let A be an object of $\mathcal{C}$ and let $\underline{A}\colon \mathcal{C} \to \mathcal{C}$ be the constant functor $\underline{A}X \triangleq A$, $\underline{A}f \triangleq 1_A$.

Proposition 4.16 *Let $t\colon TA \to A$ be a morphism. The natural transformation $[t]\colon [\underline{A}, T]$ defined by $[t]_X = t$ is an EM-law if and only if t is a T-algebra.*

Proof. Naturality of $[t]$ amounts to the obvious equation $1_A \circ t = t \circ T(1_A)$. The (η^-) axiom of $[t]$ is equivalent to the equation $[t]_X \circ \eta_{\underline{A}X}^T = \underline{A}\eta_X^T$ for all objects X, which simplifies into the unitality axiom $t \circ \eta_A^T = 1_A$. The (μ^-) axiom of $[t]$ is equivalent to the equation $\underline{A}\mu_X^T \circ [t]_{TX} \circ T[t]_X = [t]_X \circ \mu_{\underline{A}X}^T$ for all objects X, which simplifies into the associativity axiom $t \circ Tt = t \circ \mu_A^T$. □

Proposition 4.17 *Let (A, s) be an S-algebra, (A, t) be a T-algebra, and $\lambda\colon [S, T]$ be a natural transformation. Then (A, s, t) satisfies the (λ-condition) if and only if the Yang-Baxter equation holds between $[s]\colon [\underline{A}, S]$, $[t]\colon [\underline{A}, T]$ and $\lambda\colon [S, T]$.*

Proof. The Yang-Baxter equation is $[s]_{TX} \circ S[t]_X \circ \lambda_{\underline{A}X} = \underline{A}\lambda_X \circ [t]_{SX} \circ T[s]_X$ for all objects X, which simplifies into the (λ-condition) $s \circ St \circ \lambda_A = 1_A \circ t \circ Ts$. □

Therefore

- the functor λ-algebras $\to (S \circ_\lambda T)$-algebras from Theorem 4.15 is a particular case of Theorem 4.1;
- the functor λ-algebras $\to (S \bullet_\lambda T)$-algebras from Theorem 4.15 is a particular case of Theorem 4.5.

Let us illustrate *e.g.* the second point. Let $\lambda\colon [S, T]$ be a weak distributive law and (A, s, t) be a λ-algebra. According to Propositions 4.16 and 4.17, we are in the presence of a Yang-Baxter equation with one weak distributive law λ and two EM-laws $[s]$ and $[t]$. By applying a mild variation of Theorem 4.5 – consisting in forgetting the monad structure of the functor $R = \underline{A}$ – we get an EM-law $\psi = \underline{A}\pi^\lambda \circ [s]T \circ S[t] \circ \iota^\lambda \underline{A}\colon [\underline{A}, S \bullet_\lambda T]$. That law satisfies $\psi_X = s \circ St \circ \iota_A^\lambda$ for all objects X. Via Proposition 4.16, we get that $s \circ St \circ \iota_A^\lambda$ is an $(S \bullet_\lambda T)$-algebra. As expected, this procedure amounts to the action of the functor λ-algebras $\to (S \bullet_\lambda T)$-algebras, that is, $(A, s, t) \mapsto s \circ St \circ \iota_A^\lambda$.
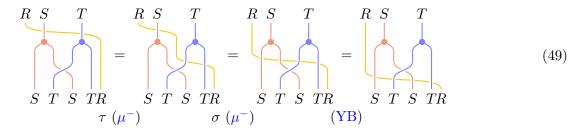
5 Discussion and Future Work

5.1 Advocating String Diagrams

One of the aims of this paper is to advocate string diagrams for proof research and communication. Let us provide two more examples that support our position.

5.1.1   Cheng's theorem

Let us begin with Cheng's result [11, Theorem 1.6], proved in [11, Appendix A]. The largest commutative diagram occurring in the proof for the initialisation case with 3 monads comprises 12 elementary commutative polygons, out of which 9 are naturality squares. Therefore, there are only 3 significant steps in the underlying proof. In an equivalent string-diagrammatic representation of the proof, these three steps are remarkably well identified:

$$\tag{49}$$

That computation (more precisely, its symmetric) has been generalised to weak distributive laws in the present paper: compare diagrams (49) and (24).

5.1.2   Winter's theorem

Another striking example arises from [38]. As already mentioned in the introduction, the proof of [38, Theorem 5] required a Prolog program to find a convenient tiling of the largest required commutative diagram, depicted in [38, Appendix A]. That diagram comprises $13 + 10 = 23$ elementary commutative polygons, out of which 15 are naturality squares. Therefore, there are only 8 significant steps in the underlying proof. An equivalent string-diagrammatic formalisation of the diagram in question is presented in Appendix C. As expected, it contains only 8 steps, and the computation can be carried out in one single line. Winter's proof research was obfuscated by what Hinze and Marsden call *bookkeeping* [20], *i.e.*, administrative operations devoid of significant mathematical content.

5.2   Conclusion

This paper generalises iteration to weak distributive laws using the graphical calculus of string diagrams. It is an additional step in developing the theory of weak distributive laws, as is our line of research in [17,18,16]. The paper revolves around one main theoretical result, split in Theorems 4.3, 4.5 and 4.6, and provides several examples. A possible extension would be to derive – in the style of Section 4.2 – a generic Yang-Baxter equation stating compatibility of *monotone* laws. Monotone laws are a certain class of laws of type $[P, T]$ and often carry significant semantic content. This direction is particularly interesting, as all known non-trivial examples of weak distributive laws are based on monotone laws. One may also ask whether the existence of a composite law entails the Yang-Baxter equation – this question has been positively solved by Winter in the case of distributive laws [38, Proposition 11].

As hinted in Section 4.3, our results easily adapt to EM-laws (and their duals, called Kleisli-laws), *i.e.*, distributive laws between a monad and a *functor*. This paves the way for applications of iterated laws in coalgebra theory – as already attempted in [38] – where both EM-laws and Kleisli-laws are ubiquitous. Another possible follow-up to the vision developed in Section 4.3 is to *weaken algebras*. Indeed, as some EM-laws turn out to be algebras, some *weak* EM-laws turn out to be so-called *semi*algebras (as defined in [15]). Dually, *coweak* Kleisli-laws probably correspond to an interesting class of free algebras. Clarifying the whole picture will most likely enhance the still underdeveloped theory of *coweak* distributive laws.

Iteration of standard distributive laws has been heavily exploited in the context of categories-as-monads [39,8,7,6]. Phrasing our results in a 2-categorical setting, as did Cheng for iterated distributive laws [11], would enable new case studies to arise in that area.

On another level, our key message is to advocate string diagrams to deal with categorical proofs in contexts where traditional equational or commutative-diagrammatic proofs become unreadable. String diagrams are often more convenient in such proofs, as they allow to take a step further by hiding part of

the complexity in the graphical formalism. Nevertheless, they also have limitations. In the main result of Cheng [11, Theorem 1.6], iteration is performed not only for three monads but for any finite number n of monads: the $n = 3$ case does only initialise an induction. Generalising the $n = 3$ result to weak laws becomes reachable using string diagrams – though some diagrams in the proof of Theorem 4.6 already have substantial size. Even with a string-diagrammatic approach, generalizing further to n monads may require tedious proofs. Our opinion is that ingenious string-diagrammatic reasoning significantly pushes back the – fuzzy and subjective – boundary between proofs that are intelligible and those that are not.

References

[1] Beck, J., *Distributive laws*, in: B. Eckmann, editor, *Seminar on Triples and Categorical Homology Theory*, pages 119–140, Springer Berlin Heidelberg, Berlin, Heidelberg (1969), ISBN 978-3-540-36091-9.
https://doi.org/10.1007/BFb0083084

[2] Bonchi, F., F. Gadducci, A. Kissinger, P. Sobocinski and F. Zanasi, *String diagram rewrite theory I: rewriting with frobenius structure*, J. ACM **69**, pages 14:1–14:58 (2022).
https://doi.org/10.1145/3502719

[3] Bonchi, F., R. Piedeleu, P. Sobocinski and F. Zanasi, *Bialgebraic foundations for the operational semantics of string diagrams*, Inf. Comput. **281**, page 104767 (2021).
https://doi.org/10.1016/j.ic.2021.104767

[4] Bonchi, F. and A. Santamaria, *Combining semilattices and semimodules*, in: S. Kiefer and C. Tasson, editors, *Foundations of Software Science and Computation Structures - 24th International Conference, FOSSACS 2021, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2021, Luxembourg City, Luxembourg, March 27 - April 1, 2021, Proceedings*, volume 12650 of *Lecture Notes in Computer Science*, pages 102–123, Springer (2021).
https://doi.org/10.1007/978-3-030-71995-1_6

[5] Bonchi, F., A. Silva and A. Sokolova, *Distribution Bisimilarity via the Power of Convex Algebras*, Logical Methods in Computer Science **Volume 17, Issue 3** (2021).
https://doi.org/10.46298/lmcs-17(3:10)2021

[6] Bonchi, F., P. Sobocinski and F. Zanasi, *Deconstructing Lawvere with distributive laws*, J. Log. Algebraic Methods Program. **95**, pages 128–146 (2018).
https://doi.org/10.1016/j.jlamp.2017.12.002

[7] Bonchi, F., P. Sobociński and F. Zanasi, *Interacting Hopf algebras*, Journal of Pure and Applied Algebra **221**, pages 144–184 (2017), ISSN 0022-4049.
https://doi.org/10.1016/j.jpaa.2016.06.002

[8] Bonchi, F. and F. Zanasi, *Bialgebraic semantics for logic programming*, Log. Methods Comput. Sci. **11** (2015).
https://doi.org/10.2168/LMCS-11(1:14)2015

[9] Böhm, G., *The weak theory of monads*, Advances in Mathematics **225**, pages 1–32 (2010), ISSN 0001-8708.
https://doi.org/10.1016/j.aim.2010.02.015

[10] Böhm, G., S. Lack and R. Street, *On the 2-categories of weak distributive laws*, Communications in Algebra **39**, pages 4567–4583 (2011).
https://doi.org/10.1080/00927872.2011.616436

[11] Cheng, E., *Iterated distributive laws*, Mathematical Proceedings of the Cambridge Philosophical Society **150**, page 459–487 (2011).
https://doi.org/10.1017/S0305004110000599

[12] Coecke, B. and A. Kissinger, *Picturing Quantum Processes: A First Course in Quantum Theory and Diagrammatic Reasoning*, Cambridge University Press (2017).
https://doi.org/10.1017/9781316219317

[13] Dahlqvist, F. and R. Neves, *Compositional semantics for new paradigms: probabilistic, hybrid and beyond*, arXiv preprint arXiv:1804.04145 (2018).
https://doi.org/10.48550/arXiv.1804.04145

[14] Dahlqvist, F., L. Parlant and A. Silva, *Layer by layer – combining monads*, in: B. Fischer and T. Uustalu, editors, *Theoretical Aspects of Computing – ICTAC 2018*, pages 153–172, Springer International Publishing, Cham (2018), ISBN 978-3-030-02508-3.
https://doi.org/10.1007/978-3-030-02508-3_9

[15] Garner, R., *The Vietoris monad and weak distributive laws*, Applied Categorical Structures **28**, pages 339–354 (2020), ISSN 0927-2852.
https://doi.org/10.1007/s10485-019-09582-w

[16] Goy, A., *On the compositionality of monads via weak distributive laws*, Ph.D. thesis, Université Paris-Saclay, Gif-sur-Yvette, France (2021).

[17] Goy, A. and D. Petrişan, *Combining probabilistic and non-deterministic choice via weak distributive laws*, in: *Proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '20, page 454–464, Association for Computing Machinery, New York, NY, USA (2020), ISBN 9781450371049.
https://doi.org/10.1145/3373718.3394795

[18] Goy, A., D. Petrişan and M. Aiguier, *Powerset-like monads weakly distribute over themselves in toposes and compact Hausdorff spaces*, in: *Proceedings of the 48th International Colloquium on Automata, Languages, and Programming*, ICALP '21, Association for Computing Machinery (2021).
https://doi.org/10.4230/LIPIcs.ICALP.2021.132

[19] Hinze, R. and D. Marsden, *Dragging proofs out of pictures*, in: S. Lindley, C. McBride, P. W. Trinder and D. Sannella, editors, *A List of Successes That Can Change the World - Essays Dedicated to Philip Wadler on the Occasion of His 60th Birthday*, volume 9600 of *Lecture Notes in Computer Science*, pages 152–168, Springer (2016).
https://doi.org/10.1007/978-3-319-30936-1_8

[20] Hinze, R. and D. Marsden, *Equational reasoning with lollipops, forks, cups, caps, snakes, and speedometers*, J. Log. Algebraic Methods Program. **85**, pages 931–951 (2016).
https://doi.org/10.1016/j.jlamp.2015.12.004

[21] Jacobs, B., *From multisets over distributions to distributions over multisets*, in: *2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–13 (2021).
https://doi.org/10.1109/LICS52264.2021.9470678

[22] Jacobs, B., A. Silva and A. Sokolova, *Trace semantics via determinization*, Journal of Computer and System Sciences **81**, pages 859–879 (2015), ISSN 0022-0000. 11th International Workshop on Coalgebraic Methods in Computer Science, CMCS 2012 (Selected Papers).
https://doi.org/10.1016/j.jcss.2014.12.005

[23] Joyal, A. and R. Street, *The geometry of tensor calculus, I*, Advances in Mathematics **88**, pages 55–112 (1991), ISSN 0001-8708.
https://doi.org/10.1016/0001-8708(91)90003-P

[24] Karamlou, A. and N. Shah, *No-go theorems for mixed distributive laws: extended abstract*, in: *CMCS short contributions*, Munich, Germany (2022).

[25] Klin, B. and J. Salamanca, *Iterated covariant powerset is not a monad*, Electronic Notes in Theoretical Computer Science **341**, pages 261–276 (2018), ISSN 1571-0661. Proceedings of the Thirty-Fourth Conference on the Mathematical Foundations of Programming Semantics (MFPS XXXIV).
https://doi.org/10.1016/j.entcs.2018.11.013

[26] Manes, E. and P. Mulry, *Monad compositions. i: General constructions and recursive distributive laws*, Theory and Applications of Categories [electronic only] **18**, pages 172–208 (2007).

[27] Manes, E. and P. Mulry, *Monad compositions ii: Kleisli strength*, Mathematical Structures in Computer Science **18**, page 613–643 (2008).
https://doi.org/10.1017/S0960129508006695

[28] Marsden, D., *Category theory using string diagrams*, arXiv preprint arXiv:1401.7220 (2014).
https://doi.org/10.48550/arXiv.1401.7220

[29] Moggi, E., *Notions of computation and monads*, Information and Computation **93**, pages 55–92 (1991), ISSN 0890-5401. Selections from 1989 IEEE Symposium on Logic in Computer Science.
https://doi.org/10.1016/0890-5401(91)90052-4

[30] Parlant, L., *Monad Composition via Preservation of Algebras*, Ph.D. thesis, UCL (University College London) (2020).
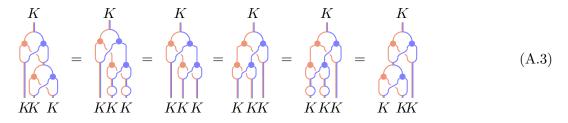https://discovery.ucl.ac.uk/id/eprint/10112228/1/Thesis_Louis_Parlant_UCL_upload.pdf

[31] Petrisan, D. and R. Sarkis, *Semialgebras and weak distributive laws*, in: A. Sokolova, editor, *Proceedings 37th Conference on Mathematical Foundations of Programming Semantics, MFPS 2021, Hybrid: Salzburg, Austria and Online, 30th August - 2nd September, 2021*, volume 351 of *EPTCS*, pages 218–241 (2021).
https://doi.org/10.4204/EPTCS.351.14

[32] Plotkin, G. and J. Power, *Notions of computation determine monads*, in: M. Nielsen and U. Engberg, editors, *Foundations of Software Science and Computation Structures*, pages 342–356, Springer Berlin Heidelberg, Berlin, Heidelberg (2002), ISBN 978-3-540-45931-6.
https://doi.org/10.1007/3-540-45931-6_24

[33] Salamanca Téllez, J., *Lattices do not distribute over powerset*, Algebra universalis **81**, page 49 (2020).
https://doi.org/10.1007/s00012-020-00680-8

[34] Silva, A., F. Bonchi, M. M. Bonsangue and J. J. M. M. Rutten, *Generalizing the powerset construction, coalgebraically*, in: K. Lodaya and M. Mahajan, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2010)*, volume 8 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 272–283, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2010), ISBN 978-3-939897-23-1, ISSN 1868-8969.
https://doi.org/10.4230/LIPIcs.FSTTCS.2010.272

[35] Street, R., *Weak distributive laws.*, Theory and Applications of Categories [electronic only] **22**, pages 313–320 (2009).
http://eudml.org/doc/233565

[36] Varacca, D. and G. Winskel, *Distributing probability over non-determinism*, Mathematical Structures in Computer Science **16**, page 87–113 (2006).
https://doi.org/10.1017/S0960129505005074

[37] Wadler, P., *Monads for functional programming*, in: J. Jeuring and E. Meijer, editors, *Advanced Functional Programming, First International Spring School on Advanced Functional Programming Techniques, Båstad, Sweden, May 24-30, 1995, Tutorial Text*, volume 925 of *Lecture Notes in Computer Science*, pages 24–52, Springer (1995).
https://doi.org/10.1007/3-540-59451-5_2

[38] Winter, J., *Product rules and distributive laws*, in: I. Hasuo, editor, *Coalgebraic Methods in Computer Science - 13th IFIP WG 1.3 International Workshop, CMCS 2016, Colocated with ETAPS 2016, Eindhoven, The Netherlands, April 2-3, 2016, Revised Selected Papers*, volume 9608 of *Lecture Notes in Computer Science*, pages 114–135, Springer (2016).
https://doi.org/10.1007/978-3-319-40370-0_8

[39] Zanasi, F., *Interacting Hopf Algebras- the Theory of Linear Systems. (Interacting Hopf Algebras - la théorie des systèmes linéaires)*, Ph.D. thesis, École normale supérieure de Lyon, France (2015).
https://tel.archives-ouvertes.fr/tel-01218015

[40] Zetzsche, S., G. van Heerdt, A. Silva and M. Sammartino, *Canonical automata via distributive law homomorphisms*, in: A. Sokolova, editor, *Proceedings 37th Conference on Mathematical Foundations of Programming Semantics, MFPS 2021, Hybrid: Salzburg, Austria and Online, 30th August - 2nd September, 2021*, volume 351 of *EPTCS*, pages 296–313 (2021).
https://doi.org/10.4204/EPTCS.351.18

[41] Zwart, M., *On the non-compositionality of monads via distributive laws*, Ph.D. thesis, Oxford (2020).
http://www.cs.ox.ac.uk/files/12453/MaaikeZwartDPhilThesis.pdf

[42] Zwart, M. and D. Marsden, *No-go theorems for distributive laws*, Log. Methods Comput. Sci. **18** (2022).
https://doi.org/10.46298/lmcs-18(1:13)2022

A   Proofs of Section 3

In this section we prove that $S \bullet_\lambda T$ is indeed a monad. To prove the first unitality axiom, *i.e.*, that $\mu^{S\bullet T} \circ \eta^{S\bullet T} K = 1_K$, we need equation (10), the equation $\iota \circ \pi = \kappa$, the retract equation $\pi \circ \iota = 1_K$, η^S unitality and the retract equation again.

$$\text{(A.1)}$$

To prove the second unitality axiom, *i.e.*, that $\mu^{S\bullet T} \circ K\eta^{S\bullet T} = 1_K$, we need equation (10), λ (η^+) axiom, unitality of both η^S and η^T, and the retract equation.

$$
\begin{array}{ccccccccc}
K & & K & & K & & K & & K
\end{array}
$$

$$(A.2)$$

As for distributive laws, it is possible to prove, using only the (μ^+) and (μ^-) axioms, that $\theta \triangleq \mu^S\mu^T \circ S\lambda T$ is associative, *i.e.*, $\theta \circ ST\theta = \theta \circ \theta ST$. To prove the associativity axiom of $S \bullet T$, *i.e.*, that $\mu^{S\bullet T} \circ K\mu^{S\bullet T} = \mu^{S\bullet T} \circ \mu^{S\bullet T}K$, we need equation (11), the retract equation, θ associativity, the retract equation again, and equation (11) again.
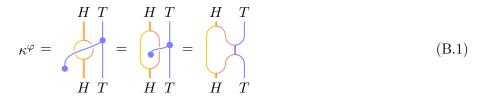
$$(A.3)$$

B   Proofs of Section 4

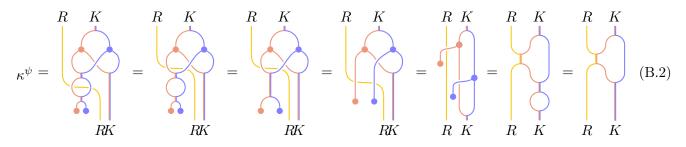In this section we provide the missing details from the proof sketch of Theorem 4.6.

B.1   *Expressions of κ^φ and κ^ψ*

First, let us justify the expressions of κ^φ and κ^ψ in equations (32) and (33). Recall that the definition of the idempotent derived from a weak distributive law $\lambda \colon [S, T]$ is given in equation (7) and in the third diagram of (12).

Computing κ^φ is easy by using the definition of the idempotent κ^φ, the (η^-) diagram for τ and the definition of the idempotent κ^λ.
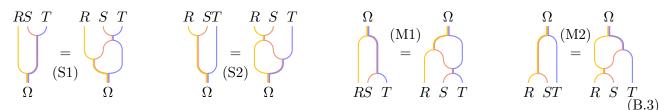
$$(B.1)$$

Computing κ^ψ is longer because it contains the composite multiplication $\mu^{S\bullet T}$. We use the definition of κ^ψ, Lemma 4.4, the retract equation, equation (10), the (η^-) axiom for τ, definitions of κ^σ and κ^λ, and the retract equation again.

$$(B.2)$$

B.2   Separate and Merge the Triple Functor

Let us prove the four equations displayed in (39), which we recall here:

(B.3)

For once we may restate and prove these properties using plain equational reasoning – they will nonetheless be applied in string-diagrammatic form in the next section. Equationally, the properties are as follows:

$$R\iota^\lambda \circ \iota^\psi = R\kappa^\lambda \circ \iota^\sigma T \circ \iota^\varphi \tag{S1}$$

$$\iota^\sigma T \circ \iota^\varphi = \kappa^\sigma T \circ R\iota^\lambda \circ \iota^\psi \tag{S2}$$

$$\pi^\psi \circ R\pi^\lambda = \pi^\varphi \circ \pi^\sigma T \circ R\kappa^\lambda \tag{M1}$$

$$\pi^\varphi \circ \pi^\sigma T = \pi^\psi \circ R\pi^\lambda \circ \kappa^\sigma T \tag{M2}$$

To prove (S1) and (M1), recall that we defined $\iota^\psi = \beta \circ \iota^\varphi$ and $\pi^\psi = \pi^\varphi \circ \alpha$, where $\alpha = \pi^\sigma T \circ R\iota^\lambda$ and $\beta = R\pi^\lambda \circ \iota^\sigma T$. Hence

$$R\iota^\lambda \circ \iota^\psi = R\iota^\lambda \circ \beta \circ \iota^\varphi \tag{B.4}$$

$$= R\iota^\lambda \circ R\pi^\lambda \circ \iota^\sigma T \circ \iota^\varphi \tag{B.5}$$

$$= R\kappa^\lambda \circ \iota^\sigma T \circ \iota^\varphi \tag{B.6}$$

and

$$\pi^\psi \circ R\pi^\lambda = \pi^\varphi \circ \alpha \circ R\pi^\lambda \tag{B.7}$$

$$= \pi^\varphi \circ \pi^\sigma T \circ R\iota^\lambda \circ R\pi^\lambda \tag{B.8}$$

$$= \pi^\varphi \circ \pi^\sigma T \circ R\kappa^\lambda \tag{B.9}$$

To prove (S2) and (M2), notice that $\iota^\varphi = \kappa^\varphi \circ \iota^\varphi = \alpha \circ \beta \circ \iota^\varphi = \alpha \circ \iota^\psi$, and similarly $\pi^\varphi = \pi^\psi \circ \beta$. Hence

$$\iota^\sigma T \circ \iota^\varphi = \iota^\sigma T \circ \alpha \circ \iota^\psi \tag{B.10}$$

$$= \iota^\sigma T \circ \pi^\sigma T \circ R\iota^\lambda \circ \iota^\psi \tag{B.11}$$

$$= \kappa^\sigma T \circ R\iota^\lambda \circ \iota^\psi \tag{B.12}$$

and

$$\pi^\varphi \circ \pi^\sigma T = \pi^\psi \circ \beta \circ \pi^\sigma T \tag{B.13}$$

$$= \pi^\psi \circ R\pi^\lambda \circ \iota^\sigma T \circ \pi^\sigma T \tag{B.14}$$

$$= \pi^\psi \circ R\pi^\lambda \circ \kappa^\sigma T \tag{B.15}$$

B.3   Units and Multiplications Coincide

At this point, we already know that the weak distributive laws $\varphi\colon [R \bullet S, T]$ and $\psi\colon [R, S \bullet T]$ generate respectively a monad $(R \bullet S) \bullet T$ and a monad $R \bullet (S \bullet T)$ whose underlying functors coincide. It remains to prove that the units and multiplications coincide. Let us begin with units.
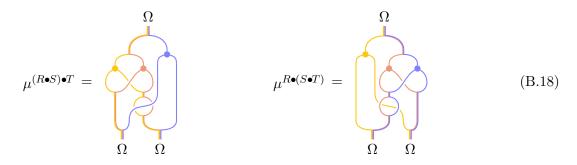
According to the general theory of weak distributive laws from Section 3, the units of the monads generated by $\varphi\colon [R \bullet S, T]$ and $\psi\colon [R, S \bullet T]$ are respectively:

$$\eta^{(R\bullet S)\bullet T} \;=\; \qquad\qquad\qquad \eta^{R\bullet(S\bullet T)} \;=\; \qquad\qquad\qquad\qquad \text{(B.16)}$$

To prove that these are equal, we just need to use equation (10) for κ^λ, and equation (M1).

$$\qquad\qquad = \qquad\qquad = \qquad\qquad\qquad\qquad\qquad \text{(B.17)}$$

The situation is more intricated for multiplications. According to the general theory of weak distributive laws from Section 3, the multiplications of the monads generated by $\varphi\colon [R \bullet S, T]$ and $\psi\colon [R, S \bullet T]$ are respectively:

$$\mu^{(R\bullet S)\bullet T} \;=\; \qquad\qquad\qquad\qquad \mu^{R\bullet(S\bullet T)} \;=\; \qquad\qquad\qquad\qquad\qquad \text{(B.18)}$$
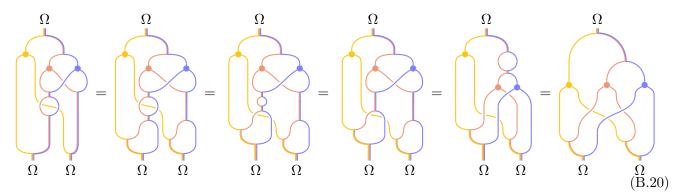
To prove that these are equal, we will manipulate separately both of them until their string-diagrammatic representations coincide. The first multiplication can be transformed as follows, using equation (M2), equation (11), the retract equation, Lemma (4.2) and the retract equation again.

$$= \; = \; = \; = \; = \qquad\qquad\qquad \text{(B.19)}$$

The second multiplication, in turn, transforms as follows, using equation (S1), Lemma (4.4), the retract

equation, equation (11) and the retract equation again.

$$\tag{B.20}$$

Hence both multiplications are equal. This achieves the proof that $(R \bullet S) \bullet T$ and $R \bullet (S \bullet T)$ coincide as monads.

C   Proofs of Section 5

In this section we restate one implication of [38, Theorem 5] and prove it using string diagrams.

Definition C.1 Let F be a functor, S and T be monads, and $S \circ T$ be their composite monad with respect to some distributive law of type $[S, T]$. A Winter-law is a natural transformation $\lambda \colon TF \Rightarrow FST$ such that

$$\lambda \circ \eta^T F = F\eta^{S \circ T} \tag{η^*}$$

$$\lambda \circ \mu^T F = F\mu^{S \circ T} \circ \lambda ST \circ T\lambda \tag{μ^*}$$

Theorem C.2 (Winter's theorem converse implication) *Let F be a functor, S and T be monads, $\lambda^0 \colon [S, T]$ be a distributive law, $\lambda^1 \colon TF \Rightarrow FST$ be a Winter-law and $\lambda^2 \colon [F, S]$ be an EM-law. Assume the following coherence axiom is satisfied:*

$$F\mu^S T \circ FS\lambda^0 \circ \lambda^1 S \circ T\lambda^2 = F\mu^S T \circ \lambda^2 ST \circ S\lambda^1 \circ \lambda^0 F \tag{coh}$$
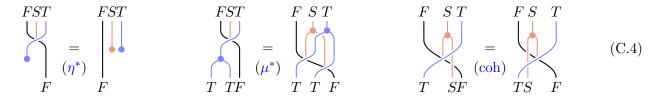
Let

$$\hat{\lambda} \triangleq STF \xrightarrow{S\lambda^1} SFST \xrightarrow{\lambda^2 ST} FSST \xrightarrow{F\mu^S T} FST \tag{C.1}$$

Then $\hat{\lambda} \colon [F, S \circ T]$ is an EM-law.

Proof. First, we express all available data using string diagrams.

$$\tag{C.2}$$

$$\tag{C.3}$$

Axioms specific to Winter's framework are displayed below:

$$
\begin{array}{ccccc}
\text{(diagram)} = \text{(diagram)} \quad (\eta^*) & & \text{(diagram)} = \text{(diagram)} \quad (\mu^*) & & \text{(diagram)} = \text{(diagram)} \quad (\text{coh})
\end{array}
\tag{C.4}
$$

By definition, the natural transformation $\hat{\lambda}$ is as follows:

$$
\hat{\lambda} \triangleq \;\text{(diagram)}
\tag{C.5}
$$

The (η^-) axiom for $\hat{\lambda}$ can be proved using $\lambda^2\ (\eta^-)$ axiom, η^S unitality and $\lambda^1\ (\eta^*)$ axiom.

$$
\text{(diagram)} = \text{(diagram)} = \text{(diagram)} = \text{(diagram)}
\tag{C.6}
$$

The (μ^-) axiom for $\hat{\lambda}$ can be proved using $\lambda^1\ (\mu^*)$ axiom, $\lambda^2\ (\mu^-)$ axiom, μ^S associativity twice, equation (coh), μ^S associativity twice again, and $\lambda^0\ (\mu^+)$ axiom.

$$
\text{(diagram)} = \text{(diagram)} = \text{(diagram)} = \text{(diagram)} = \text{(diagram)} = \text{(diagram)} = \text{(diagram)}
\tag{C.7}
$$

$\square$